

Computer Science 212

Object-Oriented Programming in Java

Lab 23

Aim: Recursive Programming.

Create a new Eclipse project for Lab23.

Write a *recursive* method that will determine if a given String is a *palindrome* (a string that reads the same forwards and backwards (except for spaces, etc.)).

For example,

```
isPalindrome("noon");
```

would return *true*.

Remember, you need a base case (the empty String is a palindrome, the *null* String is not.)

Recursive case: A String is a palindrome if the first and last letters are the same, and the String that remains after removing the first and last letters is a **palindrome**.

For example, "noon" -> "noon" -> "oo" -> ""

Hint: To break up the String, substring("noon",1,4) is "oo".

What about the spaces? Ignore them by proceeding to the next recursive call.

```
"Madam, I'm Adam"
"adam, I'm Ada "
"dam, I'm Ad "
"am, I'm A "
"m, I'm "
"m, I'm "
", I"
" I"
" I"
" "
```

You may also find these methods from the wrapper class *Character* helpful:

```
Character.isLetter(char ch)
```

```
Character.toUpperCase(Char ch)
```

Character.toUpperCase('M') is equal to Character.toUpperCase('m')

Try your program on strings that are not palindromes, as well as the following:
"noon", "Madam I'm Adam", "A man, a plan, a canal, Panama", "A Toyota"

For some other great palindromes, go to

http://www.fun-with-words.com/palin_example.html